ENDPOINT PACKET SCHEDULING SYSTEM

BACKGROUND OF THE INVENTION

[01] The present invention relates generally to a system for allowing devices connected to a network to collaborate with each other so as to transmit and receive data packets without impairment on the network.

[02] Ethernet and packet-switched Internet Protocol (IP) networks are systems for transmitting data between different points. These systems are known as "contention-based" systems. That is, all transmitters contend for network resources. All transmitters may transmit simultaneously. If they do, then network resources may be oversubscribed. When this happens, data may be delayed or lost, resulting in network impairment.

[03] As shown in FIG. 1, a conventional network comprises a plurality of Local Area Network (LAN) endpoints, such as computers connected to an Ethernet LAN. The endpoints are coupled to one or more LAN switches 102, which connect through another part of the network to one or more additional LAN endpoints 103. When endpoint 101 sends packets to endpoint 103, the packets are sent through LAN switch 102, which also handles packets from other LAN endpoints. If too many packets are simultaneously transmitted by the other endpoints to 103, LAN switch 102 may have a queue overflow, causing packets to be lost. (The word "packets" will be used to refer to datagrams in a LAN or Wide Area Network (WAN) environment. In a LAN environment, packets are sometimes called "frames." In a packet-switched WAN environment, packet-switching devices are normally referred to as "routers.").

[04] FIG. 2 illustrates the nature of the problem of dropped packets, which can occur in a LAN environment as well as a WAN environment. During periods where multiple endpoints are simultaneously transmitting packets on the network, the LAN switch 102 may become overloaded, such that some packets are discarded. This is typically caused by an internal queue in the LAN switch becoming full and thus becoming

unable to accept new packets until the outgoing packets have been removed from the queue. This creates a problem in that transmitting endpoints cannot be guaranteed that their packets will arrive, necessitating other solutions such as the use of guaranteed-delivery protocols such as Transmission Control Protocol (TCP). Such solutions may be inappropriate for streaming video or other real-time applications, which cannot wait for retransmission of packets.

[05] Another solution, proposed in my previous U.S. application serial number 10/663,378, involves scheduling the transmission of packets by the originating endpoints based on an empirical evaluation of network congestion conditions. A transmission interval is partitioned into discrete frames and subframes, and each endpoint schedules packets for delivery during time slots in the subframes corresponding to empirically determined conditions of minimal network congestion. That scheme relies on the existence of multiple priority levels for packets in the network, such that packets can be sent using a lower-level "discovery" priority level to perform the empirical determination without affecting the higher-priority data traffic.

[06] Some networks and devices cannot support multiple priority levels for data packets. For example, some packet switches may support only one level of packet priority (i.e., two queues: one for prioritized packets and another for non-prioritized packets), making such a scheme difficult to implement. Consequently, the present invention proposes a different solution to using a contention-based network, such as an Ethernet LAN or a WAN packet switching network, to transmit time-sensitive data such as streaming video.

SUMMARY OF THE INVENTION

[07] The invention provides a method for transmitting packets in a network by scheduling them for delivery based on an agreement between the transmitting node and the receiving node as to a delivery schedule.

[08] A transmitting node transmits a query to the intended receiving node. The receiving node responds with a reception map indicating what transmission time slots have already been allocated by other transmitting nodes (or, alternatively, what transmission time slots are available). The transmitting node then proposes a transmission map to the receiving node, taking into account any time slots previously allocated to other transmitting nodes. The receiving node either accepts the proposed transmission map or proposes an alternate transmission map. Upon agreement between the nodes, the transmitting node begins transmitting according to the proposed transmission map, and the receiving node incorporates the proposed transmission map into its allocation tables. Because the proposed delivery schedule has been agreed to between the two endpoints, uncoordinated contention that might otherwise overflow network switches near the endpoints is avoided. Because the schedule is determined by the two endpoints, no network arbiter is needed to coordinate among network resources.

[09] In another variation, a transmitting node transmits a bandwidth requirement to an intended recipient node, indicating the bandwidth it requires to support a proposed transmission (e.g., streaming video packets). The intended recipient node, after evaluating time slots previously allocated to other transmitters, responds with a proposed delivery schedule indicating time slots during which the transmitter should transmit packets in order to avoid contention with other previously scheduled packets while maintaining the necessary bandwidth for the transmitter. The transmitter thereafter transmits packets according to the proposed delivery schedule.

[10] In yet another variation, a transmitting node transmits a proposed delivery schedule to an intended recipient, indicating time slots corresponding to times during which it proposes to transmit packets. The intended recipient either agrees to the proposed delivery schedule, or proposes an alternate delivery schedule that takes into account the transmitter's bandwidth requirements. Upon agreement between the nodes, transmission occurs according to the agreed-upon delivery schedule. The schedule can be released at the end of the transmission.

BRIEF DESCRIPTION OF THE DRAWINGS

[11] FIG. 1 shows the problem of bursty packets creating an overflow condition at a packet switch, leading to packet loss.

[12] FIG. 2 shows how network congestion can cause packet loss where two sets of endpoints share a common network resource under bursty conditions.

[13] FIG. 3 shows one method for coordinating a delivery schedule for transmissions between a transmitting node and an intended recipient node.

[14] FIG. 4 shows a second method for coordinating a delivery schedule for transmissions between a transmitting node and an intended recipient node.

[15] FIG. 5 shows a third method for coordinating a delivery schedule for transmissions between a transmitting node and an intended recipient node.

[16] FIG. 6 shows a frame structure in which a transmission interval can be decomposed into a master frame; subframes; and secondary subframes, for a 10 megabit per second link.

[17] FIG. 7 shows one possible reception map for a given transmission interval.

[18] FIG. 8 shows a scheme for synchronizing delivery schedules among network nodes.

[19] FIG. 9 shows how network congestion is avoided through the use of the inventive principles, leading to more efficient scheduling of packets in the network.

DETAILED DESCRIPTION OF THE INVENTION

[20] FIG. 3 shows one method for carrying out the principles of the invention. Before describing this method, it is useful to explain how packets are scheduled for delivery over the network between nodes according to the invention.

[21] Turning briefly to FIG. 6, a transmission interval is partitioned into units and (optionally) subunits of time during which data packets can be transmitted. In the example of FIG. 6, an arbitrary transmission interval one hundred milliseconds (a master frame) can be decomposed into subframes each of 10 millisecond duration, and each subframe can be further decomposed into secondary subframes each of 1 millisecond duration. Each secondary subframe is in turn divided into time slots of 100 microsecond duration. Therefore, a period of 100 milliseconds would comprise 1,000 slots of 100 microseconds duration. According to one variation of the invention, the delivery time period for each unit of transmission bandwidth to a receiving node is decomposed using a scheme such as that shown in FIG. 6, and packets are assigned for transmission to time slots according to this schedule. This scheme is analogous to time-division multiplexing (TDM) in networks.

[22] Depending on the packet size and underlying network bandwidth, some varying fraction of each time slot would be actually used to transmit a packet. Assuming a packet size of 125 bytes (1,000 bits) and a 10BaseT Ethernet operating at 10 mbps, a single 100-microsecond time slot would be used to transmit each packet. Assuming a packet size of 1,500 bytes, twelve of the 100-microsecond intervals would be consumed by each packet transmission.

[23] According to one variation of the invention, the scheduled delivery scheme applies to prioritized packets in the network; other non-prioritized packets are not included in this scheme. Therefore, in a system that supports only priority traffic and non-priority traffic, the scheduled delivery scheme would be applied to all priority traffic, and ad-hoc network traffic would continue to be delivered on a nonpriority basis. In other words, all priority traffic would be delivered before any nonpriority traffic is delivered.

[24] The delivery schedule of FIG. 6 is intended to be illustrative only; other time period schemes can be used. For example, it is not necessary to decompose a transmission interval into subframes as illustrated; instead, an arbitrary interval can be divided up into 100-microsecond time slots each of which can be allocated to a particular transmitting node. Other time periods could of course be used, and the invention is not intended to be limited to any particular time slot scheme. The delivery schedule can be derived from a clock such as provided by a Global Positioning System (GPS). The means by which time slots are synchronized in the network is discussed in more detail below.

[25] Suppose that a transmitting node needs to support a voice connection over the network. For a single voice-over-IP connection, a bandwidth of 64 kilobits per second might be needed. Assuming a packet size of 80 bytes or 640 bits, this would mean that 100 packets per second must be transmitted, which works out to (on average) one packet every 10 milliseconds. In the example of FIG. 6, this would mean transmitting a packet during at least one of the time slots in every tenth secondary subframe at the bottom of the figure. (Each time slot corresponds to 100 microseconds, so on average, one packet every 10 milliseconds would be needed, or one packet every ten secondary subframes).

[26] Returning to FIG. 3, in step 301, a transmitting node sends a query to an intended receiving node in the network for a reception map.

[27] A reception map (see FIG. 7) is a data structure indicating time slots that have already been allocated to other transmitters for reception by the receiving node (or, alternatively, time slots that have not yet been allocated, or, alternatively, time slots that are candidates for transmission). More generally, a reception map is a data structure that indicates -- in one form or another -- time slots during which transmission to the intended receiving node would not conflict with other transmitters. Although there are many ways of representing such a map, one approach is to use a bitmap wherein each bit corresponds to one time slot, and a "1" indicates that the time slot has been allocated to a transmitting node, and a "0" indicates that the time slot has

not yet been allocated. FIG. 7 thus represents 25 time slots of a delivery schedule, and certain time slots (indicated by an "x" in FIG. 7) have already been allocated to other transmitters. If a 100-millisecond delivery interval were divided into 100-microsecond time slots, there would be 1,000 bits in the reception map. This map could be larger, for higher bandwidths. For instance, for a 100 megabit per second link, the map could have 10,000 bits, etc., to represent the same throughput per slot.

[28] In step 302, the intended receiving node responds with a reception map such as that shown in FIG. 7, indicating which time slots have already been allocated to other transmitters. If this were the first transmitter to transmit to that receiving node, the reception map would be empty. It is of course also possible that time slots could have been previously allocated to the same transmitter to support an earlier transmission (i.e., the same transmitter needs to establish a second connection to the same recipient).

[29] In step 303, the transmitter sends a proposed transmission map to the intended receiving node. The proposed transmission map preferably takes into account the allocated time slots received from the intended receiving node, so that previously allocated time slots are avoided. The transmitter allocates enough time slots to support the required bandwidth of the transmission while avoiding previously allocated time slots.

[30] Suppose that a virtual connection is to be established between two nodes on the network to support a telephone voice connection. A voice-over-IP connection may require 64 kilobits per second transfer rate using 80-byte packet payloads (not including packet headers). A video stream would typically impose higher bandwidth requirements on the network. On an Ethernet LAN, each packet would comprise up to 1,500 bytes, which (at 10BaseT rates) could be transmitted in approximately 12 100-microsecond periods or slots. A voice-over-IP connection could be established by transmitting one 80-byte payload packet every 10 milliseconds.

[31] In step 304, the intended recipient reviews the proposed transmission map and agrees to it, or proposes an alternate transmission map. For example, if the intended recipient had allocated some of the proposed time slots to another transmitter during the time that the transmitter was negotiating for bandwidth, the newly proposed delivery schedule might present a conflict. In that situation, the intended recipient might propose an alternate map that maintained the bandwidth requirements of the transmitter.

[32] In step 305, the transmitter repeatedly transmits to the intended recipient according to the agreed delivery schedule. To support a voice-over-IP connection, for example, the transmitter could transmit an 80-byte packet every 10 milliseconds. For a streaming video connection, the transmitter could transmit at a more frequent rate. Finally, in step 306 the receiver's map is deallocated when the transmitter no longer needs to transmit.

[33] Note that for two-way communication, two separate connections must be established: one for node A transmitting to node B, and another connection for node B transmitting to node A. Although the inventive principles will be described with respect to a one-way transmission, it should be understood that the same steps would be repeated at the other endpoint where a two-way connection is desired.

[34] FIG. 4 shows an alternative method for carrying out the inventive principles. Beginning in step 401, the transmitter sends a bandwidth requirement to the intended recipient. For example, the transmitter may dictate a packet size and bandwidth, and the intended recipient could determine which slots should be allocated to support that bandwidth. In step 402, the intended recipient responds with a proposed transmission map that takes into account previously allocated time slots.

[35] In step 403, the transmitter agrees to the proposed transmission map, causing the intended receiver to "lock in" the agreed time slots (this step could be omitted), and in step 404 the transmitter transmits packets according to the agreed-upon schedule.

Finally, in step 405 the transmission map is deallocated upon termination of the connection.

[36] FIG. 5 shows another variation of the inventive method. In step 501, the transmitting node sends a proposed transmission map to the intended recipient. In step 502, the intended recipient either agrees to the proposed transmission map (if it is compatible with any previously-allocated maps) or proposes an alternative map that meets the transmitter's bandwidth requirements, which can be inferred from the proposed transmission map. For example, if the transmitter had proposed transmitting in time slots 1, 11, 21, 31, 41, and so forth, it would be evident that the transmitter needed to transmit once every tenth time slot. If the requested slots were not available, the intended recipient could instead propose slots 2, 12, 22, 32, and so forth.

[37] In step 503, the transmitter transmits packets according to the agreed-upon delivery schedule, and in step 504 the transmission map is deallocated upon termination of the transmission.

[38] In another variation, a transmitter may request bandwidth (e.g., one 1000-byte packet every 10 milliseconds) and the receiver responds with a placement message (e.g., start it at the 75th 100-microsecond slot). The receiver could also respond with multiple alternatives (e.g., start it at the 75th, the 111th, or the 376th time slot). The transmitter would respond with the time slot that it intended to use (e.g., the 111th), and begin transmission. This variation is intended to be within the scope of sending "transmission maps" and "reception maps" as those terms are used herein.

[39] In order for each transmitter and receiver to agree on a delivery schedule, it is desirable or necessary to develop and maintain some time synchronization between the nodes. FIG. 8 shows one possible approach for synchronizing delivery schedules among nodes in a network.

[40] As shown in FIG. 8, the network comprises various endpoints connected through a switch 802. According to one variation of the invention, a clock source 804 (e.g., a

GPS-derived clock) is coupled through an electrical wire 805 to each network node participating in the scheduled delivery scheme. The clock source generates pulses that are transmitted to each node and used as the basis for the delivery schedule. Each node may comprise a timer card or other mechanism (e.g., an interrupt-driven operating system) that is able to use the timing signals to establish a common reference frame. This means for synchronizing may therefore comprise a physical wire (separate and apart from the network) over which a synchronization signal is transmitted to each node. It may further comprise a hardware card and/or software in each node to detect and decode the synchronization signal.

[41] The clock pulses may comprise a pulse according to an agreed-upon interval (e.g., one second) that is used by each node to generate time slots that are synchronized to the beginning of the pulses. Alternatively, the clock source may generate a high-frequency signal that is then divided down into time slots by each node. Other approaches are of course possible. As yet another alternative, each node may contain its own clock source that is synchronized (via GPS or other means) to a common reference signal, such as a radio signal transmitted by the U.S. Government. Wire 805 may comprise a coaxial cable or other means of connecting the clock source to the nodes. In one variation, the connection is of a short enough distance (hundreds of feet) so that transmission effects and delays are avoided. Any of these means for synchronizing may be used independently of the others.

[42] Another way or means of synchronizing time slots and delivery schedules among the nodes is to have one node periodically transmit (e.g., via multicast) a synchronization packet on the node on the network. Each node would receive the packet and use it to synchronize an internal clock for reference purposes. As an alternative to the multicast approach, one network node can be configured to individually send synchronization packets to each participating network node, taking into account the stagger delay involved in such transmission. For example, a synchronization node would transmit a synchronization packet to a first node on the network, then send the same packet to a second node on the network, which would be received later by the

second node. The difference in time could be quantified and used to correct back to a common reference point. Other approaches are of course possible.

[43] FIG. 9 illustrates how practicing the inventive principles can reduce congestion by more efficiently scheduling data packets between transmitters and receivers. As shown in FIG. 9, because each transmitting node schedules packets for delivery during times that do not conflict with those transmitted by other nodes, no packets are lost.

[44] While the invention has been described with respect to specific examples including presently preferred modes of carrying out the invention, those skilled in the art will appreciate that there are numerous variations and permutations of the above described systems and techniques that fall within the spirit and scope of the invention as set forth in the appended claims. Any of the method steps described herein can be implemented in computer software and stored on computer-readable medium for execution in a general-purpose or special-purpose computer, and such computer-readable media is included within the scope of the intended invention. The invention extends to not only the method but also to computer nodes programmed to carry out the inventive principles. Numbering associated with process steps in the claims is for convenience only and should not be read to imply any particular ordering or sequence.